

```

-module(game).

-compile(export_all).

-record(score, {score = {0,0} :: {integer(), integer()}
               | deuce | {a,integer()} | {win,integer
()}}).

%% rules: 0, 15, 30, 40 ( game | deuce ), (a | game | deuce)
%%
%%

new() ->
    #score{}.

score(Player, Rec) when Player==1; Player==2 ->
    Next = fun(0) -> 15;
            (15) -> 30;
            (30) -> 40
        end,
    NewScore = case Score = Rec#score.score of
                deuce      -> {a, Player};
                {a, Player} -> {win, Player};
                {a, _}     -> deuce;
                _ ->
                    case {Player, Score} of
                        {1, {40,_}} -> {win, 1};
                        {2, {_, 40}} -> {win, 2};
                        {1, {30,40}} -> deuce;
                        {2, {40,30}} -> deuce;
                        _ ->
                            setelement(
                                Player, Score, Next(element(Player,
Score)))
                    end
                end,
    Rec#score{score = NewScore}.

```